CENTRE FOR
SYSTEMS
SCIENCE

Simon Fraser University, Burnaby, B.C., Canada V5A 1S6 / (604) 291-4704

Reasoning with Truth Ascriptions, Self-Reference and Embedded Sentences in First-Order Logic

Robert F. Hadley

CSS-IS TR 90-04

April 1990

92-05737/

Reasoning with Truth Ascriptions, Self-Reference and Embedded Sentences in First-Order Logic

by

Robert F. Hadley

School of Computing Science
Simon Fraser University
Burnaby, Canada V5A 1S6
Tel. 604-291-4488

Abstract

A first-order logic, $\mathcal{T}$-FOL, is presented, which *at once* sanctions inferences involving truth ascriptions, while *also* permitting self-reference and arbitrary nesting of sentences within propositional attitudes. $\mathcal{T}$-FOL includes a distinguished truth predicate and special inference rules which permit inferences involving truth-ascriptions to be drawn in a natural, intuitive fashion, while avoiding explicit contradictions of the kind noted by Tarski (1936). *Explicitly* self-referential sentences (such as the famous 'Liar' sentence) cannot be represented in $\mathcal{T}$-FOL, but standard, *implicit* self-reference is included. The adoption of a (Fregean-based) context-sensitive semantics allows self-reference, and arbitrary nesting of sentences to be achieved without the encumberance of quotation marks and concatenation devices. Within $\mathcal{T}$-FOL, the objects of which truth, knowledge, and belief are *predicated* are *intensions*, rather than particular sentences. This approach enables a many-to-one mapping between particular sentences and the beliefs they express. Applications of $\mathcal{T}$-FOL to inferences involving truth are described. The soundness and completeness of $\mathcal{T}$-FOL have been established and are referenced within.

Copyright © 1989 Robert F. Hadley

Introduction

Truth ascriptions, and inferences involving truth ascriptions are an important part of human dialogue. Likewise, *implicitly self-referential* statements, and nested epistemic assertions play an important, if less frequent role. Consider the following examples:

(a) Mary: "Bill said he wasn't interested in dating Sally."
John: "Well, Bill is not always truthful about his romantic life."

(b) Sue: "Nobody's infallible; not all my beliefs are true. On the other hand, when I tell you something about New York, it's very likely to be true.

(c) Bill: "I called because I didn't know whether you knew I would be at the party".

John's assertion in example (a) involves a denial of truth to some of Bill's statements. This truth-denial may very well lead Mary to alter her attitude towards Bill, and to adopt a skeptical attitude towards several of Bill's other statements. Example (b) presents two sentences, the first of which could well be (implicitly) self-referential, since the speaker presumably counts that very sentence among her beliefs. The second sentence involves a truth ascription which could, if accepted, lead the listener to accept several of Sue's statements about New York. In (c) we are presented with an epistemic proposition (involving knowledge) nested within another epistemic assertion.

Examples such as the above form an integral part of human cognition. Yet, surprisingly little attention has been devoted to the development of a logic which *at once* sanctions inferences involving truth ascriptions while *also* permitting self-reference and arbitrary nesting of epistemic sentences within other epistemic sentences. A notable prior attempt to integrate these multiple aspects of human cognition within a formal setting is Perlis' (1985) first-order treatment of truth. A minor disadvantage of Perlis' approach is his use of nested quotation marks and string-concatenation functions to achieve a first-order language. In section 3, more serious shortcomings are described, involving aspects of Perlis' theory of truth.

The primary goal of this paper is to present an alternative to Perlis' logic of truth, which would both support truth inferences in the presence of truth ascriptions, and permit the representation of nested propositional attitudes. Like Perlis, I retain a first-order representation language, but arbitrary embedding of propositions is achieved without resort to quotation and concatenation functions. The language of first-order logic (FOL) is chosen for reasons which are presented in section 2. Truth inferences are sanctioned by the addition of two rules of inference to standard FOL. These rules involve a distinguished truth-predicate ('T'), and the resulting augmented logic is dubbed $\mathcal{T}$-FOL. Within $\mathcal{T}$-FOL, the objects of which truth, knowledge, and belief are

predicated are *intensions*, rather than particular sentences. This approach enables a many-to-one mapping between particular sentences and the beliefs they express.

To summarize, the novel scientific contributions of $\mathcal{T}$-FOL are these:

- The inclusion of truth-specific rules of inference within $\mathcal{T}$-FOL allows inferences involving truth-ascriptions to be drawn in a natural, intuitive fashion, while avoiding explicit contradictions of the kind noted by Tarski (1936). *Explicitly* self-referential sentences (such as the famous 'Liar' sentence) cannot be represented in $\mathcal{T}$-FOL, but standard, *implicit* self-reference is included.
- $\mathcal{T}$-FOL's intensionally-based semantics ensures a many-to-one mapping between particular sentences and particular beliefs; thereby avoiding the *narrowly* syntactic approach adopted by Perlis (1985) and Haas (1986) in their first-order epistemic languages (cf. Hadley, 1989b).
- *Context-sensitive* semantic conventions render embedded predicates, quantifiers, and connectives as functions -- thereby permitting a straightforward compositional semantics to be assigned to embedded sentences while retaining a first-order syntax. Embedded sentences may appear as arguments to *any* predicate, nested at any level.
- Soundness and Completeness for $\mathcal{T}$-FOL are established.

Finally, applications of $\mathcal{T}$-FOL to inferences involving truth are described in section 6.

1. The Intensional Approach

The intensional approach is founded upon the premise, widespread among philosophers of language, that belief and knowledge must be understood as a relation between an agent and an abstract *intension* (proposition or thought). Recent work in epistemic modeling, which adopts an intensional stance, includes that of Wilks and Ballim (1987), who are concerned primarily with the *generation* of nested beliefs (especially atypical beliefs). By contrast, our concern is not with the *generation* of nested beliefs, but with adequate methods of *representing* nested epistemic states (including beliefs involving complex nesting of quantifiers), and with the logic of truth ascriptions. As will emerge, the concept of an 'intension' is fundamental to both these concerns. In order to better understand the senses of 'intension' at issue, let us briefly review some of the history of this concept.

The thesis that a declarative sentence expresses an abstract sense or intension, whose structure is a *function* both of the sentence's syntax and of the meaning of its parts, is a view which dates back at least to Frege. Modern philosophical descendents of Frege, including Montague (1970) and Lewis (1976), have considerably refined and modified Frege's view, but Frege's underlying idea that the sense (intension) of a sentence is determined by a compositional process, involving both syntax and meaning functions, persists. The precise definition of 'intension' varies from theory to theory, but a sentence's intension is usually taken to be a *function* which, given a possible world as argument, returns the truth value of the sentence in that possible world.

One notorious difficulty with intensional semantics is that when intensions are merely construed

as functions in the set-theoretic sense, then every tautology, ϕ, turns out to have the same intension (or meaning), since each valid ϕ must be assigned the value 'true' in each possible world. (Recall that ϕ's intension is a function from possible worlds to truth values). Thus, when intensions are regarded *merely* as functions, intensions are too coarse-grained to serve as meanings or "thought contents". However, one can obtain finer-grained intensions by including structural (syntactic) information in the definition of an intension. Syntactic methods for varying the granularity of intensions are described in (Lewis, 1976; Hadley, 1988).

Apart from the intensional theories mentioned above, other *compositional* theories of sentence meaning exist, and each has its own degree of granularity. Cresswell [1985] proposes a theory of sentence meaning with an attractive degree of granularity, as does Gawron [1986]. For the present, we may remain neutral among these competing theories. We shall simply identify intensions with sentence meanings, and note that our choice of granularity for intensions could vary according to the intended application of $\mathcal{T}$-FOL. What is relevant for present purposes is that each of the aforementioned theories proposes a *canonical* system for representing meanings or intensions, such that *all sentences which are synonymous* (according to the semantic theory) *are mapped onto a single canonical representation.* Moreover, each of the semantic theories presupposes the existence of a *principled* set of rules for translating (or compiling) an expression into its canonical meaning representation. In computational applications it is useful to identify intensions (which are abstractions) with these canonical representations, since the latter are manipulable syntactic objects. However, since a variety of distinct, synonymous sentences can each map onto a single representation of an intension, intensional approaches are *not narrowly* syntactic, i.e., they need not posit a 1-1 mapping between sentences and beliefs.

2. The Language of $\mathcal{T}$-FOL

The language of $\mathcal{T}$-FOL is designed to permit the representation of deductive arguments involving *nested* epistemic propositions and truth ascriptions in FOL. Although $\mathcal{T}$-FOL does not contain a formal model of belief, it is intended to be used in conjunction with the first-order *theory* of belief and *belief-change* described in (Hadley, 1990). This first-order theory axiomatizes facts about change of belief which can only be expressed in a language at least as rich as FOL. One such fact could be paraphrased thus: "If agent X believes S at time *t1*, and *t3* is a later time than *t1*, and there is no time between *t1* and *t3* when agent X retracts S or calls S into question, then X still believes S at *t3*." (Here, 'retracts' may be interpreted broadly to include spontaneous 'forgetting'.) When this paraphrase is rendered in FOL, the resulting sentence does not belong to a decidable subset of FOL. Nevertheless, agents who reason about each other's beliefs (under circumstances where either agent might retract a belief in order to maintain consistency) need to be aware of facts such as the above. $\mathcal{T}$-FOL permits agents to represent such facts to themselves, and this is a major reason why the language of FOL is

employed here, despite its computational intractability.[1] Moreover, we may eventually wish to allow epistemic assertions to involve parameters not only for agent, object of belief, and time, but for *degrees of confidence* in the belief. The use of FOL allows us to add new parameters to particular predicates in a straightforward fashion. On the other hand, the representation of *nested* epistemic propositions in FOL is not straightforward.

Perlis (1985) describes a method, involving quotation, unquotation, and concatenation of expressions, which permits the representation of nested propositions in an extension of FOL. Haas (1986) presents a quotation-based scheme which is, in many respects, analogous to that of Perlis. However, both the methods of Perlis and Haas become unwieldy at levels of nesting greater than one. By contrast, the language of $\mathcal{T}$-FOL involves no quotation mechanism. $\mathcal{T}$-FOL relies instead on the *composition* of *functions* to ensure both that the objects of belief are intensions, and that inferences are not unduly encumbered by an plethora of quotation marks. For example, consider the following sentence which involves self-reference within an embedded sentence.

(1) Mary knows that John believes that none of John's beliefs are true.

(Although John's belief here is certainly puzzling, similar beliefs are held by people who suscribe to certain forms of philosophical relativism.)

In $\mathcal{T}$-FOL (1) would be represented as

(2) $Knows(mary, Believes(john, \neg\exists\, x\{\, Believes(john, x) \,\&\, True(x)\,\}\,))$

Now, at first glance (2) appears not to be a sentence of first-order logic (FOL). However, as we shall see, if we allow the role of predicates, quantifiers, and truth connectives to be *context sensitive*, so that they express functions in any embedded context, then (2) becomes a sentence of FOL. The underlying motivation for this approach is not merely to simplify our system of representation and inference (though we do obtain those benefits). Rather, we are attempting to embody Frege's thesis that the denotation of a sentence (or wff) varies according to its surrounding context. Frege maintained that sentences which occur in embedded epistemic contexts (as in (1)) denote their *sense*, and not their truth value. As previously mentioned, this *sense* (intension) is formed through the composition of functions. Additional restrictions may be

---

[1]Our overriding concern is not with finding a computationally tractable logic, but with providing a representation system and logic that approaches the richness of human systems. Moreover, it is by no means clear that the present superiority of humans (over computers) in certain domains is attributable to humans using tractable logics (as opposed to a combined use of mental models, efficient indexing schemes, and topically organized knowledge, say). We know, for example, that many forms of mathematical reasoning are intractable, and the resultant theorems are not expressible in a tractable logic. Yet, human mathematicians surpass the best AI programs in this domain. In addition, many planning problems appear insoluable, except by means of intractable search processes. If artificial agents are to engage in *hard* planning of this kind, it seems clear that they will require intractable logics. Also, if these agents are to formulate beliefs *about* how their plans were derived, it is difficult to see how these beliefs could be *represented* except in an intractable logic.

placed upon intensions to obtain differing degrees of granularity, but all intensional theories relevant to our discussion posit that intensions are formed through the composition of functions. In some respects, what we are proposing is similar to an *n*-order intensional logic (where *n* is unbounded). However, all formulae which we shall need to represent are expressible in FOL. We shall be extending standard FOL by the inclusion of two rules of inference (which concern the predicate 'true'), but these do not affect the representation *language* (Hereafter, '$\mathcal{L}$' shall denote the language of $\mathcal{T}$-FOL.)

In order to implement Frege's thesis concerning the context-sensitivity of denotations, we adopt the following notational conventions: the predicates *T* (for true), *K* (for knowledge), and *B* (for belief) are distinguished predicates, and always appear in upper case; other predicates, constants, and variables are written in lower case. Officially, the truth-functional connectives, '$\neg$', '$\rightarrow$', etc., are *prefix* connectives whose arguments are written within parentheses. Thus, we write $\rightarrow (\forall x\,(\,pred1(x)\,),\ \exists y\,(\,pred2(y)\,))$. Also, any time a quantifier is introduced, the top-level formula within its scope is enclosed in parentheses, as in $\exists x\,(\,pred2(x)\,)$. In practice, we shall omit parentheses where the scope is unambiguous, and use connectives in their customary infix position. Our official notation is intended merely to stress that quantifiers and connectives may be viewed as functors. Of course, predicates are commonly regarded as functors by modern logicians; thus '*red(x)*' is usually called a propositional function.

Now, when a predication (say, *red(x)*) is embedded in an epistemic context, the predicate involved (*red*) expresses a function, *f1*, whose parameter is *x*, and whose *value* is a proposition whose identity varies according to *x*. Thus, when *x* is assigned the value *a*, the function *f1* returns, *but does not evaluate*, a proposition which is the standard, non-embedded intension of '*red(a)*'. This intension is a function, $f2_a$, which assigns different truth values to '*red(a)*' in different possible worlds. Analogously, when a truth-connective, (say, '$\rightarrow$') appears in an embedded context (as in *K(john,* $\rightarrow$*(red(x),round(x))))*, it expresses a function, *f3*, which takes as arguments whatever propositions (intensions) are returned by *red(x)* and *round(x)*, and *f3* returns a new function, *f4*, which is simply the standard truth function attached to $\rightarrow$*(red(x), round(x))* in FOL. Of course, *f3* cannot actually return a given intension until a value is assigned to *x*. When a value (say, *a*) is assigned to *x*, *f3* returns ($f4_a$) the intension of $\rightarrow$*(red(a), round(a))*. Again, *f3* merely returns, but does not evaluate $f4_a$.

The behaviour of quantifiers in embedded contexts, although similar in some respects to the above, differs in important respects. To see this, let *f5* be the function expressed by '$\forall x$' in an embedded context, and let *g1* be the function expressed by '*round*'. Then consider the general form of *K(john,* $\forall x$ *{*$\rightarrow$*(red(x), round(x))}, time–c)* when *embedded* quantifiers, connectives, and predicates are replaced by the function names we have introduced. Performing the replacement gives us:

(3) *K(john, f5{f3(f1(x), g1(x))}, time–c)*

Given that *f5*, *f3*, *f1* and *g1* are all functions, *f5* (which is expressed by '$\forall x$' only in embedded

contexts) ultimately returns an intension by means of straightforward composition of functions. Now, at first glance, it may appear that occurrences of 'x' in $f5\{f3(f1(x), g1(x))\}$ are free occurrences, since *f5* is just the outer-most function occurring in the atomic formula (3). This appearance is illusory, however. To be sure, occurrences of 'x' in *f3(f1(x), g1(x))* are free, but when this latter expression is supplied as an argument to *f5*, the *value returned* by *f5* is the ordinary *non-embedded* intension of $\forall x\{\rightarrow(red(x), round(x))\}$, which does not contain free occurrences of 'x'. This is because *f5* is a function expressed by an *embedded* quantifier of 'x'. As such, *f5 expects* to be passed a (propositional) function involving free occurrences of 'x'. So, although *f3* cannot return a value until 'x' is bound to a constant, *f5* can return a value as soon as it is passed the open function *f3*(....x....). (If we were working in LISP, *f5* would be an NLAMBDA function, because *f5* does not attempt to evaluate its argument before using it.) What *f5* returns is another function (*f6*), but *f6* is simply the sense of a closed, quantified sentence of $\mathcal{L}$.[2] Now, if the argument supplied to *f5* was a function containing free occurrences of another variable 'y', as well as 'x', then the value returned by *f5* would still be an open function -- but only open with respect to 'y' -- occurrences of 'x' would not be free.

By contrast, consider the sentence, $\exists x(B(john, blue(x), time\text{–}c))$. The predicate '*blue*' in this embedded context expresses a function which returns particular propositions for given values of 'x'. Since no *embedded* quantifier binds occurrences of 'x' here, closure of *B(john,....)* requires the prefixing of a quantifier (in this case, $\exists x$).

3. Truth in $\mathcal{T}$-FOL and in Perlis

Hopefully, it is now clear that the functions we have chosen to assign to predicates and logical operators -- *when they occur in embedded contexts* -- are such that when a first-order sentence, ψ, is nested inside another wff, ψ *returns* (but does not evaluate) the function which *constitutes* the ordinary, non-embedded sense of ψ. For this reason, we may justify the addition of two rules of inference to our first-order theory, given the convention just described. These rules allow us to move an expression into or out of an embedded context of the following form:

(4) $T(\phi)$.

Informally, we may construe T as the "truth" predicate. Thus, (4) would assert that the *intension* denoted by ϕ is 'true'. A more formal interpretation for T is provided in section 4. At this stage, it is important to note that we do not attempt to *explicitly* define the truth predicate within the *language* of $\mathcal{T}$-FOL. However, the following two rules of inference express the *logic* of this predicate, and they are included in our extension of FOL.

T1: ϕ may be derived from T(ϕ), provided ϕ has the typographic form of a (closed) sentence of $\mathcal{L}$.

---

[2]Although *f6* is not evaluated in this embedded context, *f6* would return truth just in case, for every value assignable to x in the present possible world, *f3* yields a proposition which *would* evaluate to true.

T2: T(ϕ) may be derived from ϕ.

Note that T1 allows us to "extract" *functional* expressions from a truth context, under certain conditions, and T2 allows us to *embed* a sentence in a truth context. Now, there are circumstances in which we wish to assert a sentence such as *'T(c)'*, where *'c'* is an individual constant denoting some intension. In such cases, T1 does *not* allow us to derive *c* as an assertion, since *c* is not a sentence of $\mathcal{L}$.

Insofar as T1 and T2 allow us to move formulae into and out of a truth context, their joint effect bears some resemblance to Perlis' axiom schema:

(5) True("A") ≡ A*

where A is a meta-variable ranging over sentences of the language, and * is an operater which ensures that (5) is equivalent to

(6) True("A") ≡ A

in those cases where 'A' does not contain the predicate 'True'. In effect, the * operator is a device which *blocks* the derivation of True("*L*") from *L* in cases where *L* ≡ ¬*True*("*L*"), i.e., in cases where *L* is the famous "Liar" sentence. In such cases, Perlis allows that *L* and ¬*True*("*L*") may both be *proven*, while *True*("*L*") may not be derived. Thus, he avoids explicit contradiction, but in the process he adopts a sense of 'truth' which may do violence to our ordinary concept of truth. For, it seems counter-intuitive to say that we are *sometimes* perfectly entitled *on semantic grounds* to *assert* a sentence (in this case *L*), but we are not entitled on those occasions to say the sentence is true. Kripke's view, which many would share, is that a correct theory of truth ought to preserve the following common usage: we say that a sentence is true in *exactly* those situations where we are semantically entitled to simply assert the sentence. (Note that $\mathcal{T}$-FOL's rules, T1 and T2, preserve this common usage. *Whenever* we have justifiably asserted a sentence B, T2 allows us to infer "True B". T1 sanctions the reverse inference for *any* sentence B.)

Apart from the above difficulty, it seems clear that Perlis' use of the * operator (in (5)) cannot, in general, accomplish what he wants it to. To see this, we need to have in mind Kripke's notion of a *grounded* sentence. In essence, a grounded sentence is one whose truth evaluation "bottoms out", either immediately or ultimately, in the evaluation of sentences which do not ascribe truth or falsity to other sentences. By contrast, the truth evaluation of an *ungrounded* sentence does not "bottom out", but triggers an endless series of further attempts to evaluate the truth of itself and/or other sentences. (For more on how truth evaluation can "ultimately bottom out", see Hadley, 1989.)

Now, Perlis clearly intends that the "star" operator will block the derivation of *True("A")* from *A* when *A* is not grounded.[3] In Perlis' schema, the "star" operator works in a purely syntactic

---

[3]Indeed, Perlis must hold this if (5) is to prevent 'True' from being ascribed to the subtle kinds of paradoxical sentences which Kripke discusses (cf. Kripke, 1975, and Perlis, 1985, pp. 310-311).

fashion; in essence, it rewrites its argument by replacing occurrences of $\neg True$("...") by occurrences of $True$("$\neg$(...)"). But the question whether a sentence is grounded *sometimes* depends upon purely empirical considerations, and cannot reliably be determined by syntactic means. This was a major conclusion of Kripke's paper. Given Kripke's conclusion (which Perlis does not dispute), it follows that the star operator cannot do both the following things: (a) always block the derivation of *True("A")* from *A*, when *A* is ungrounded, (b) always permit (as Perlis claims) the derivation of a sentence having the form *True(...)*, from *A*, when *A* is grounded. Since the same sentence can be grounded is some possible worlds, and not in others, no purely syntactic device can reliably distinguish between valid and invalid derivations of *True(...)* from ungrounded *A*. As it happens, schema (5) works reliably only for *explicitly* self-referential sentences, such as the Liar.

4. Paradox and Truth in $\mathcal{T}$-FOL

Let us now consider whether the paradoxes of self-reference can arise in $\mathcal{T}$-FOL. To begin, we observe that the explicit liar paradox cannot be formulated in $\mathcal{T}$-FOL. To be sure, we can represent *instances* of the following within the language of $\mathcal{T}$-FOL ($\mathcal{L}$):

(7) $\phi \equiv \neg T(\phi)$

by replacing ϕ with any sentence of $\mathcal{L}$, but all *instances* of (7) are simply *false*, not paradoxical. (Within $\mathcal{L}$, only a well-formed, closed formula could be substituted for ϕ in (7). Because there is no convention within $\mathcal{L}$ by which a closed formula could *denote* itself, it is not possible to find a value for ϕ which renders (7) paradoxical in $\mathcal{L}$.) From our standpoint, (7) is merely an absurd meta-schema, and is not expressible in $\mathcal{L}$. Moreover, because rules T1 and T2 are *sound* (cf. section 5), we cannot use rules T1 and T2 to derive an instance of (7) from a consistent premise set. Now, *if* we adopted as a premise, $\neg T(c)$, and *if* the constant c were allowed to name the intension of this premise, then we could demonstrate in our meta-language that $\neg T(c)$ is semantically paradoxical. However, we could not derive a corresponding contradiction within $\mathcal{T}$-FOL, because

(8) $c \equiv \neg T(c)$

is not even well-formed. Moreover, even at the meta-level, we cannot coherently suppose that a constant names the intension of an expression (ψ) in which that constant occurs, because the intension (of ψ) can *exist* only if we have *already* assigned an intension to the given constant. This follows from the compositionality of intensions.

Now, although there appears to be no way in which an explicit liar paradox can be expressed in $\mathcal{L}$, this fact does not preclude the more subtle kinds of paradox described by Kripke (1975). These depend upon exceedingly rare (and arguably, farfetched) empirical conditions arising when each of a pair of sentences refers to the truth of the other. It is doubtful whether the required empirical conditions will arise in fact. However, we should consider whether the mere possibility that semantic paradoxes will arise renders $\mathcal{T}$-FOL defective.

To begin, let us note that Kripke clearly demonstrates [1975] that *any* language rich enough to

permit the *ascription* of truth to sentences is vulnerable to paradox. He further demonstrates that this vulnerability in no way presupposes syntactic or semantic ill-formedness. Now, it is true that nearly all logics current in AI escape this vulnerability, but only because they do not sanction inferences involving truth ascriptions. For many specialized applications, one can safely ignore the logic of truth ascriptions, but if we are ever to have an adequate model of *human cognition*, we will need to represent and reason with propositions such as, "What Mary believes is true".

Since we are ultimately concerned with adequate cognitive models, we have included a truth predicate. Thus, paradoxical sentences may arise. However, Kripke [1975] has shown that the inclusion of such sentences need not render a first-order language inconsistent. Kripke proposes a complex theory of truth according to which paradoxical sentences are simply not assigned truth values. The underlying intuition of Kripke's theory is that any ascription of truth to a sentence involves a *presupposition* that the sentence is grounded. On the rare occasions when the presupposition of groundedness is unsatisfied no truth value is assigned. Kripke goes on to argue (following Kleene, 1952) that the existence of sentences without truth values does not require that we abandon any of the usual laws of logic. For example, since "$p \vee \neg p$" still holds for all cases when p *has* a truth value, we may retain the law. When p lacks a truth value, the disjunction is *not false*; rather it also lacks a truth value.

The question now arises, must the interpretation of T, the truth predicate in $\mathcal{T}$-FOL, involve considerations of *groundedness*? In the final analysis, the answer is yes. On the face of it, the interpretation of T is straightforward. We merely assign as the extension of T the set of all intensions which evaluate to 'true', where 'true' is an object in the domain of $\mathcal{L}$. However, upon reflection, we note that the relevant "set of true intensions" includes not only intensions of sentences taken from an agent's representation language (whatever that may be), but also intensions belonging to some sentences of $\mathcal{L}$. Now, some of the latter intensions arise from sentences containing the predicate T. Thus, it becomes problematic that *these* latter intensions should enter into the definition of the *extension* of T (because, from an ontological standpoint, the intension and extension of a term arise together). The problem is to prevent a vicious circularity from entering into our specification of the "set of true intensions" (which we denote by 'S').

Our solution, in its essentials, derives from Kripke (1975). We define the membership of S *recursively*, as follows: (In what follows, assume that ϕ is a sentence in the domain of $\mathcal{L}$. The domain of $\mathcal{L}$ includes sentences and intensions belonging to the representation languages of the agents concerned, and also *all* sentences and intensions belonging to $\mathcal{L}$ itself.)

(i) If ϕ contains no occurrences of T, and *if* the intension of ϕ returns 'true' on the current interpretation of ϕ, then the intension of ϕ is in S.

(ii) Otherwise, if ϕ is a sentence of $\mathcal{L}$ having the form $T(\psi)$, then the intension of ϕ is that function f which returns 'true' iff ψ is both grounded and the intension of ψ evaluates to 'true'. If f returns 'true' on the current interpretation of $\mathcal{L}$, then $f \in S$.

(iii) Otherwise, ϕ is a molecular or quantified sentence of $\mathcal{L}$, and contains some occurrence of T. The intension of ϕ (again, call it f) is formed according to the standard compositional rules of Tarskian semantics, and $f \in S$ iff f returns 'true' on the current interpretation of $\mathcal{L}$. Since the subexpressions of ϕ are recursively evaluated for truth in the normal Tarskian fashion, the evaluation of f ultimately reduces to conditions (i) and (ii), which are straightforward.

Given the above definition, the membership of S is fully specified. This, in turn, provides a determinate extension for the predicate T. The fact that T's extension is recursively defined is no more problematic than if we had assigned the natural numbers as T's extension. As it happens, however, the chosen interpretation assures the soundness of rules of inference, T1 and T2, as we shall soon see.

5. Formal Properties of the Model

We now consider some of the formal properties of $\mathcal{T}$-FOL. First, note that the axioms of $\mathcal{T}$-FOL, with minor syntactic alterations, may be represented in any standard first-order predicate calculus, known to be sound and complete. In what follows, 'PC' shall denote those formalizations of *pure* first-order predicate logic, which do *not* contain the propositional calculus as a subset.[4]

Definition. Let $\mathcal{T}$ be a first-order logic formed by adding to PC the two inference rules T1 and T2. Further assume that whatever Tarskian interpretation is assigned to $\mathcal{T}$, the *predicate* T is assigned the truth interpretation described in the previous section.

Proposition 1. $\mathcal{T}$ is sound.

Proof. The inference rules of $\mathcal{T}$ are exactly those of PC combined with the truth rules T1 and T2. Since the inference rules of standard PC are known to be sound, we need only demonstrate that T1 and T2 are also truth preserving. Because both these rules are explicitly concerned with truth, the proof is straightforward. In essence, T1 allows us to derive ϕ from $T(\phi)$, provided ϕ is a closed wff of $\mathcal{L}$. Now, suppose that $T(\phi)$ is true under an interpretation I, and that ϕ is a closed wff of $\mathcal{L}$. By definition of $\mathcal{T}$, I assigns the predicate T the *truth* interpretation. Furthermore, our definition of a *truth* interpretation (section 2.4), requires that $T(\phi)$ can be true only if ϕ has a true intension. But, because ϕ is a wff of $\mathcal{L}$, its intension will be true only if ϕ is true under I. Thus, T1 is truth preserving.

To see that T2 is truth preserving, we reverse the foregoing argument. T2 allows us to derive $T(\phi)$ from ϕ. Suppose that ϕ is a sentence of $\mathcal{L}$ which is true under I (as above). Then the intension of ϕ must evaluate to 'true'. Consequently, the intension of ϕ is in the extension of T. Therefore, $T(\phi)$ is true under I.

Proposition 2. $\mathcal{T}$ is complete.

The proof of completeness for $\mathcal{T}$ is non-trivial. Due to space limitations, the proof is ommitted

---

[4]We wish to exclude those formalizations of FOL which permit *constants* to stand alone as wffs of the language. Thus, a sentence such as $P \equiv T(P)$ does not occur in any logic denoted here by 'PC'.

here. It is provided in the full technical report (Hadley, 1990).

6. Applications of the Truth Predicate

I have previously noted the importance of the concept of *truth* in human dialogue. I also claimed that rules T1 and T2 capture important aspects of commonsense reasoning about truth. To support this last claim, I now consider ways in which T1 and T2 may be useful in acquiring and transmitting inductive beliefs. Suppose a robot, Bill, hears a number of Mary's remarks (call them *r1*, *r2*, *r3*, etc.) concerning the habits of waterfowl. And suppose that Bill remembers these remarks, and later has an opportunity to verify them. In time, Bill discovers that each of Mary's remarks describes what is the case. That is, Bill observes that *r1*, *r2*, *r3*.... Using rule T1, Bill infers *T(r1)* from *r1*, *T(r2)* from *r2*, etc. As time passes, Bill has a growing list of observations of the form:

$$asserts(mary, p) \wedge concerns(p, waterfowl) \wedge T(p).$$

Eventually, Bill makes an inductive leap and postulates

(9) $\forall p \{ [asserts(mary, p) \wedge concerns(p, waterfowl)] \rightarrow T(p) \}$

The use of rule T1 has been a necessary step in the formulation of Bill's generalization. Now, suppose Bill conveys (9) (in essence) to Jane, who has a growing interest in waterfowl. If Jane believes Bill, she may now regard Mary as an authority on waterfowl. When Jane is told that Mary asserted *q* about waterfowl, Jane may use (9) to infer *T(q)*, and then use T2 to infer *q*. In this manner, *q* may enter Jane's belief base, which in turn permits Jane to act on *q*, (e.g., Jane now looks for woodducks in the first week of November). Although the preceding example by no means exhausts the ways in which we use the concept of truth, it does show that T1 and T2 have a useful role to play in everyday reasoning.

In passing, we should note that in addition to the applications just outlined, $\mathcal{T}$-FOL has interesting applications in the realm of inter-agent reasoning. Space does not permit a discussion of these applications here, but they are fully described in conjunction with the first-order theory of belief presented in (Hadley, 1990).

Summary

Since the publication of Tarski's (1936) renowned work on truth in formal languages, a commonly held misconception has arisen, namely, that the logic of our everyday concept of *truth* cannot be captured in a consistent first-order logic. In the foregoing, we have argued that the logic of our 'truth' concept can indeed be captured in FOL -- at least in the sense which is relevant to the Tarski controversy (and given particular assumptions about the *semantics* of embedded sentences). The importance of the concept of truth in human dialogue has been underscored through examples. In addition, the context-sensitive semantics of $\mathcal{T}$-FOL supplies a method by which *arbitrary* sentences of the logic can be used as arguments to arbitrary predicates, without the awkwardness of nested quotation and concatenation functions. In this

respect, $\mathcal{T}$-FOL is significantly more general than modal epistemic logics which permit embedding only in the context of distinguished modal operators (such as K and B). The language of $\mathcal{T}$-FOL also permits self-reference and quantification to occur at any level of embedding. Moreover, although $\mathcal{T}$-FOL provides a representation language for expressing embedded epistemic statements, it does not impose a particular *theory* of belief upon the user. An empirically based theory of belief is provided as a companion to $\mathcal{T}$-FOL in (Hadley, 1990), but alternative models may readily be expressed using $\mathcal{T}$-FOL. Finally, the soundness and completeness of $\mathcal{T}$-FOL as a logic have been proven.

Acknowledgements

The author gratefully acknowledges the support of the Natural Sciences and Engineering Council of Canada (Grant No. A0899) during the period of this research.

I wish to thank Raymond Jennings, and Charles Morgan for their helpful comments on an earlier version of this paper.

References

Cresswell, M.J., (1985) *Structured meanings: the semantics of propositional attitudes*, MIT Press, Cambridge, MA.

Gawron, J.M. (1986) "Types, contents, and semantic objects", *Linguistics and Philosophy*, Vol. 9, No. 4, pp. 427-476.

Haas, A., (1986) "A syntactic theory of belief and action", *Artificial Intelligence*, Vol. 28, 245-292.

Hadley, R.F. (1990) "An intension-based model of nested epistemic processes", Technical Report CSS-IS TR 90-01, Centre for Systems Science, Simon Fraser University, Burnaby, B.C. Canada, V5A 1S6.

Hadley, R.F. (1989) "A default-oriented theory of procedural semantics", *Cognitive Science*, Vol. 13, pp. 107-137.

Hadley, R.F. (1989b) "The many uses of 'belief' in AI", *Proceedings of the 11th Annual Conference of the Cognitive Science Society*, August, Ann Arbor, pp. 115-122.

Hadley, R.F. (1988) "Logical omniscience, semantics and models of belief", *Computational Intelligence*, Vol. 4, pp. 17-30.

Kleene, S.C., (1952) *Introduction to meta-mathematics*, Wolters-Noordhoff Publishing and North-Holland Publishing, New York.

Kripke, S., (1975) "Outline of a theory of truth", *Journal of Philosophy*, Vol. 72, pp. 690-716.

Lewis, D. (1976) "General semantics", in *Montague Grammar*, (ed.) Partee, B., Academic Press, New York.

Montague, R., (1970) "Universal grammar", *Theoria* 36.

Perlis, D., (1985) "Languages with self-reference I: foundations", *Artificial Intelligence*, Vol. 25, pp. 301-322.

Tarski, A. (1936) "Der warheitsbegriff in den formalisierten sprachen", *Studia Philos.*, Vol. 1,

pp. 261-405.

Wilks, Y. & Ballim, A. (1987) "Multiple agents and the heuristic ascriptions of belief", Proceedings of IJCAI, Milan, pp. 118-124.

Centre for Systems Science

• Simon Fraser University established the **Centre for Systems Science** as a provincial centre for advanced study. The objective of CSS is twofold: coordinate and combine advanced systems research at SFU, and encourage application of research results within industry in order to further the technological development of British Columbia.

• CSS has assembled three research groups to streamline its efforts. The orientation of these groups is as follows: intelligent systems, microelectronics, and computer and communications systems. Because of shared research interests, this latter group merges with LCCR (**Laboratory for Computer and Communications Research**) for the publication of technical reports.

• CSS's three groups involve over 80 skilled faculty researchers principally from SFU's Schools of Engineering Science and Computing Science, but also draw substantially on Mathematics, Physics, Geography, Philosophy, Linguistics, Psychology, Communication, and Kinesiology.